Information about the new server such as its name, network address, and telephone number, along with which database of keys on the CD-ROM disk is assigned to the new server needs to be given to the user's access program. For example, if 200 keys are already assigned to existing servers, the 201st key might be assigned to a new server. This information could be included in either encrypted or unencrypted form on an update floppy disk or other portable medium, posted on a bulletin board or server, including on any or all of the existing servers, or undated automatically by the remote terminal access program during a subsequent communication session. Such information may not be particularly private, as it is typically the same for all users being granted access to the new server.

The user's access program would typically store the update information for the new servers in a small file on the user's hard-drives. If the users have a writable CD-ROM drive, the information could be added to the CD-ROM disk key. If the information about each server comprise no more than 50 characters, a 10 kilobyte disk file could contain information on at least 100 new servers. A file a few megabytes in size would allow a short description of each server.

Eventually, the new servers would be included on undated CD-ROM disk keys distributed to all users.

Informational, transactional, and promotional databases are all of commercial interest. Access can be controlled, verified, or tabulated by the CD-ROM key. In addition, the individual CD-ROM disks may contain all or portions of these databases. The portions of the databases that change infrequently might be encoded on the users' CD-ROM disks and updated when new disks are produced, whereas variable portions might typically be stored on the server.

The host computer can be programmed to grant different access privileges to different users. For example, in a corporate network, the C.E.O.'s CD-ROM key would grant him access to all information on the host computer, while a clerk s disk might only grant access to a data entry program. Similarly, in a consumer application, different consumers might have different credit limits. The requisite privilege or privilege level might either be encoded on the CD-ROM or, preferably, would be included in a database on the host computer.

The CD-ROM key of the invention may contain both unencrypted and encrypted versions of one or more identification keys. The encryption is done before or as the disk is imprinted using a key and encryption method unknown to the user and using encryption means that are ideally unknown to the user. For user authentication purposes, the host computer, which has the key, would be programmed to demand both the unencrypted version of the identification key and the encrypted version of the key. The host computer then would be programmed to decrypt the encrypted version of the key and compare it with the unencrypted version. If the two keys are the same, then the user identification key is almost certainly a valid key. For example, if the encryption were the inverse of a long-key public-key encryption, the public key would be held by the host computer only (and the inverse or private key would be held by the disk maker only). An intruder would have to generate a counterfeit identification with the corresponding encrypted version, which would require the inverse or private key. Obtaining the key would be virtually impossible, even if the would-be counterfeiter obtained huge numbers of different user disks. And since even the server does not have the private key, cracking the server would not allow a counterfeiter to make new counterfeit user identification keys. Accordingly, the counterfeiting of valid user ID numbers can be eliminated.

A further security measure would be to append the encrypted version of the identification key to the unencrypted version to form a single longer key. Alternatively, the final key might comprise two different encrypted versions of the unencrypted key. Alternatively, the final key might be a function of both the unencrypted version and of a parity, hash, encryption function, or other function of the unencrypted version.

In addition, in certain applications, provisional initiation of the transaction upon receipt of a valid ID by the host computer might be permitted, but the transaction is completed only when the ID is verified in the server's database. This arrangement improves response time for the user and reduces the speed requirements on the storage means. For example, a credit card transaction could be started upon receipt of a valid ID but not completed until after the ID has been checked with the database and approved.

Unlike a human user, the computer does not make mistakes in entering an identification key. Accordingly, unless line disruption is indicated, the preferred software implementation will disconnect the user after only one attempt using any invalid CD-ROM identification key. This allows speedy rejection of attempts by hackers or other transgressors and avoids tying up the system with their illicit attempts. By disconnecting after one attempt, hackers cannot rapidly try multiple identification keys.

If this option is implemented, it is also preferable to not allow log-on if line disruption is indicated; else a hacker could counterfeit a parity failure or the like to allow multiple access key attempts. It may also be preferable to disconnect the user if more than, for example, three line disruptions are indicated during attempts to log-on.

The host computer's database of user identification keys is well protected against attempts to steal or copy it. Nevertheless, it is advantageous to protect against attempts to steal or copy the server's database of user identification keys or user access keys and thereby counterfeit the users' unique CD-ROMs. Accordingly, the server database of a preferred implementation of the invention contains an encrypted or otherwise altered version of the user identification keys. The server of the invention employs a trap-door authentication algorithm to compare the user ID or access key recovered from the incoming data stream with the altered version in the server's own database for that user's claimed identity. The trap-door authentication algorithm authenticates the user if and only if the encrypted identification key in the server's database represents the same identification key as the one embedded or encrypted in the incoming data stream. The trap-door authentication algorithm is impractical to be used to recover the actual identification key from the encrypted key in the host computer's database. Since the server database does not contain the actual identification keys, and the trap-door authentication function is of no help in recovering them, mere possession of the host computer's database is not sufficient to recover the identification keys. Thus, stealing or copying the host computer s database of identification keys will not allow a thief to counterfeit the users' unique CD-ROM key access disks and thus will not allow the thief to access the system as a legitimate user.

One such trapdoor authentication algorithm is implemented as follows. When preparing the users' CD-ROMs and the database for the host computer, the users' unique identification keys are encrypted with a difficult-to-decrypt